Deep Learning-based Human Implantable Nano Molecular Communications

Bon-Hong Koo, Ho Joong Kim, Jang-Yeon Kwon, Chan-Byoung Chae*
School of Integrated Technology
Yonsei University, Korea
Email: {harpeng7675, meerschaum, jangyeon, cbchae}@yonsei.ac.kr

***Abstract*—In this paper, we propose a novel nano-molecular communication system, including a nano receiver design and detection strategies. We show how machine intelligence can be incorporated into the practical implementation of nano communications. We introduce a testbed employing a biosensor chip as a receiver. The chip is made to be sufficiently small to be implanted under the human skin with no harm while detecting concentrations of glucose molecules over time. Molecules are released by a transmitter, to convey information through a thin pipe. For this configuration, the channel model is unknown, and the sensor dynamics can differ with according to the manufacturing process. Therefore, it is more desirable to find a universal strategy than using closed-form channel expressions so that it can be less sensitive to the channel and sensor variation. Learning-based approaches are likely to solve the problem. Therefore, in this paper, we suggest detection strategies with and without machine learning. We first describe our intuitions of nanomachine design from observations, and we show how the learning-based techniques can benefit the system by reducing the design burden and enhancing the accuracy of data detection. The study concludes by showing sample results of real data transmission.**

***Index Terms*—Molecular communication, nano machine, chip fabrication, machine intelligence.**

I. Introduction

There are limits to conventional communication techniques' abilities to serve nanoscale devices, so researchers have studied an alternative using molecules as information carriers [1]. One of the main target scenarios for adopting nano communications is *in vivo* applications for medical purposes. It follows that the molecule becomes a messenger candidate, as it has better biocompatibility and energy efficiency than the electromagnetic wave.

Molecular communication begins from its first concept drawing [2]. A theoretical framework based on diffusion equations has been established to define and describe the baseline dynamics. The diffusive molecules stay longer in the channel and yield extended memory and a lower data rate than a radio-based system. Hence, the main design issues were to mitigate interference and boost the speed. The solutions are proposed conceptually, supported mathematically and simulated virtually for verification.

The Monte Carlo-based N3Sim simulator has been a favorable method of verification, though it is yet far from practice [3]. There has been proposed a world-first testbed that transfers information via molecules [4]. In [5], it is shown that a design upgrade based on theory achieves the results with an accordingly upgraded prototype.

Regarding *in vivo* nanodevices, some notable prior work has investigated pipe-like channels, which is a closer imitation of a blood vessel [6], [7]. Rogers(2016) studied vessel-like communication environments by applying microfluidics and verified the performance with virtual simulations. Nariman(2017) devised a novel testbed employing acid and base molecules. The other remarkable work [8] also implemented a prototype employing nano particles while they are activated by electromagnetic controls.

In our system model, we propose a nano communication testbed that models communications between nanodevices inside the human body as shown in Fig. 1. In a real-world scenarios, a tiny chip that serves as a glucose-detecting sensor is implanted just beneath the human skin. In practice, the information is encoded by glucose molecules and propagates through a flexible pipe with a nanoscale diameter. The sensor is produced manually and planted inside a chunk of diaphanous protein.

In this paper, we impart the difficulties of practical implementations based on observations and show that, with natural detection methods, the issues are in fact surmountable. Then we propose machine intelligence-aided solutions to overcome the difficulties with better performance. The robust control of manufacturing is observed to be hardly achievable, so we claim that, for general use, the learning-based detections are more feasible. In contrast with prior work, the solutions here are empirically proposed, technically supported by machine learning, and experimentally verified.

The three main contributions of this paper are as follows. First, we propose a nano transceiver, where a transmitter can control the release timing and the amount of the glucose molecules, and where a receiver senses any consequently varying concentrations. Second, we design baseline detection algorithms, investigating several nonlinear factors resulting from the machine, such as mechanical instability and chemical reaction statistics. Finally, we suggest machine learning-based detections that outperform baseline algorithms.

The rest of this paper is organized as follows. In Section II, we introduce our nanomachine design, including the fabricated nano transceiver, a whole testbed and discussions from a communication perspective. In Section III, we propose detection algorithms from observed intuitions. We introduce

978-1-7281-5089-5/20/$31.00 ©2020 IEEE

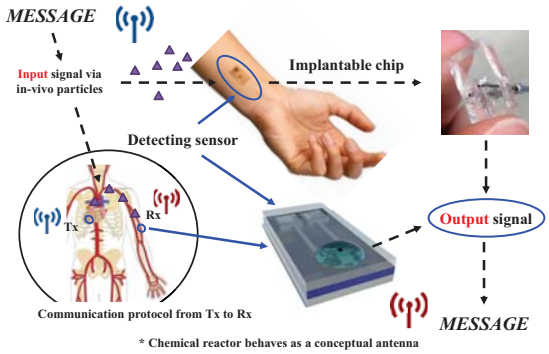

Fig. 1. The concept diagram of the proposed system. A skin implantable chip is employed to receive data from the controlled glucose molecules.

machine intelligence-aided enhancements. We give results and conclusions in Sections IV and V, respectively.

II. Nanomachine for Communications

In this section, we introduce our nano communication machine. The machine requires external devices for controlling the input and converting the output signal. Nevertheless, we refer to it as a nanomachine, since the communication functions are embedded on the nanoscale. To the best of our knowledge, it is the first communication system constructed on the nanoscale, due to the absence of an operable nanosensor at the receiver side. We first briefly describe the chip-production process with its desirable specifications. We then explain the whole part of the testbed shown in Fig. II and the analogy to a target scenario. We conclude the section by sharing technical challenges from a communication perspective.

A. Nano Receiver Fabrication

A nano biosensor needs to meet several conditions, especially if it is to be employed for *in vivo* communication purposes. The first condition is biocompatibility: the materials and by-products should be safe for the human body. Biosensors are sensitive and shorter-lived than large sensors, and the second condition should be reproducibility; they should be easy and cheap to mass produce. The last point we consider is the chip performance: how fast does it respond, how accurate is its value, and how sensitive is the sensor? Since these immediately affect communication performance.

The InGaZnO (IGZO)-based Electrolyte-Gated Field-Effect Transistors (EGFETs) is chosen as the proper sensor material. First of all, glucose itself is a biocompatible material that is omnipresent in the body. EGFETs meet the design concerns of highly sensitive, label-free detection performances and low operational voltage. In terms of field-effect mobility and a large on-off ratio, IGZO outperforms alternative materials from preliminary works such as graphene, poly-Si-nanowires, and In_2O_3-nanosheet [9]. These advantages enable the sensor to have higher sensitivity and noise robustness. Moreover, the IGZO-based circuits and sensors are known to have prominent

Table 1
The set of parameters and choice of values.

	Term	Value and Variations
t_s	Symbol interval	0.5, 1, 2, 3 (s)
t_w	Injection duration	0.25, 0.5, 1, 2 (s)
v	Drift rate	500, 1000 (μl/min)
r	Glucose injectino rate	1000, 2000, 4000 (μl/min)

biocompatibility and have thus attracted great interest from the integrated healthcare system [10].

To complete the transistor as a glucose sensor, we immobilize 10mg/ml of glucose oxidase (GOx) at the surface of the transistor. The glucose sensor mechanism is as follows [11]:

$$Glucose+H_2O+O_2+GOx \rightarrow Gluconic\,acid+H_2O_2+H^+. \quad (1)$$

The hydrogen ions produced by the right-hand side of the equatino above cause a pH variation in the medium. The transistors in our manufacturing lab detected a medium pH value ranging from 3 to 9, with a sensitivity of 52.8±1.3 mV/pH and with response times of less than 5s. The detection range of the glucose concentration is 0.05mM to 2mM. Note that the pH variation occurs locally because of a chemical reaction between minute glucose and the oxidase on the chip surface, which does not affect the entire system.

B. Communication Testbed Implementation

The transmitter part of the communication system is a syringe containing a glucose aqueous solution as an information carrier. The system is reactable to small amount of molecules. Hence, it is essential to maintain precise control of the release of the materials. A syringe pump is capable of controlling the flow rate to be 1.26 $\mathrm{pl/min}$ and the time step to be 30 μs, within 0.5 % accuracy. By coding one bit per a transmission, the achievable data rate is limited to 2 bps.

The syringe is connected to the medium by a rubber pipe with a diameter of $10\mu\mathrm{m}$, which is analogous to the human vein. The rate and the duration are the control parameters. We adjust the parameters according to the pilot performance of the produced transistors.

There is one more pump and a syringe filled with a saline solution, which is deployed to supply constant fluid into the medium, thus imitating blood flow. The flow rate for the medium drift is 500 $\mu\mathrm{l/min}$. Two separate rubber pipes meet at a confluence and constitute a medium. The other end of the medium consists of a chunk of protein, which corresponds to the human skin where the transistor is implanted.

The probing chamber protects the system from external vibrations and assists the measurements by holding the material and monitoring a sensor surface with a magnifier. The chamber measures the signal with three pin probes and sends the signal to the operation program. The program decides the measurement frequency and visualizes the output. The maximum frequency determined by the program is 8 Hz.

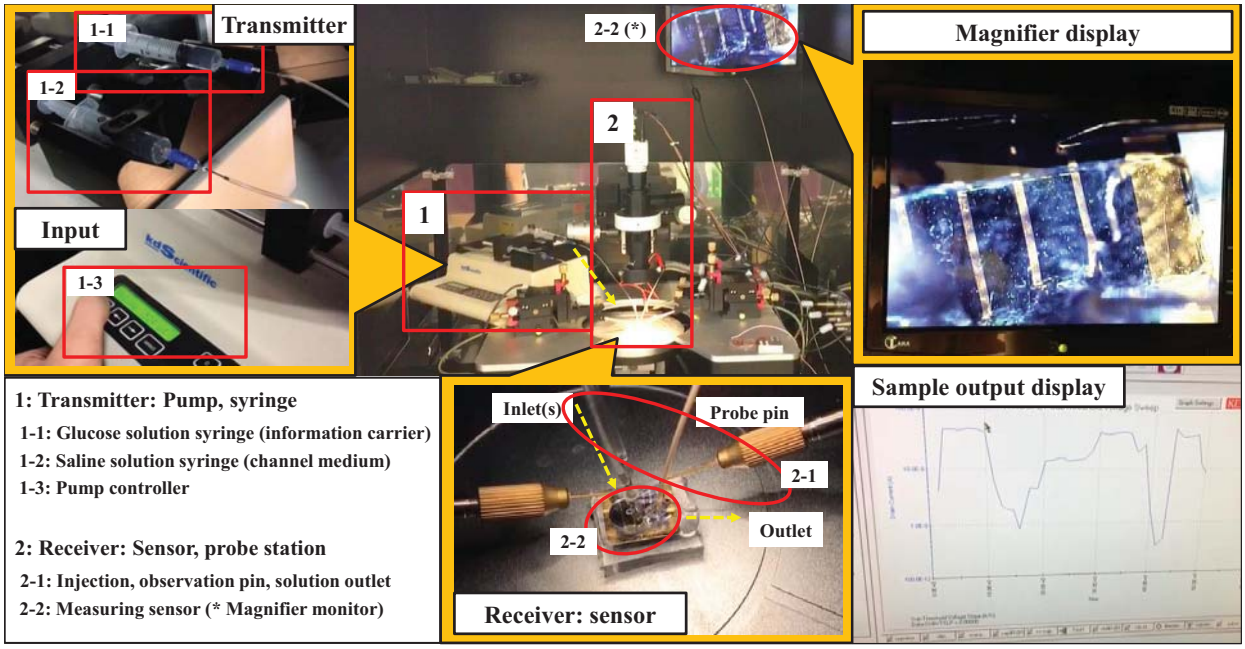

Fig. 2. The entire body of the nanomachine setup for the molecular communications employing the fabricated nano receiver.

The information is given as a sequence of binary bits, which is modulated by On-Off Keying (OOK); releasing molecules in a given time slot conveys bit-1, otherwise bit-0. Due to the mechanical limitations, the achievable data rate is bounded by 2 bps. As shown in Table 1, we try four different data rates. For simplicity, the pulse shapes are unified for every release, where the pulse shape is defined by the power and the duration of pressing the glucose syringe.

A single bit sequence consists of 50 to 100 bits, and the system randomly generates 20 different unbiased sequences as test data. With variations of system parameters, we obtained measurements of 2k bits transmission. In the next section, we will present our design intuitions from the measurements.

C. Challenges from a Communication Perspective

For a communication task, the biosensors' problem is their short lifespan, as the sensing mechanism is irreversible. This implies that the receiver is not rechargeable and the battery life is short. Note that the chip is designed to be cheap enough for mass production. Thus, in a use-case scenario, the biosensors can be replaced easily, rather than there being a necessity to increase their capacity. The problem is that the output level is highly dependent on chip conditions, and the fabrication randomness is hard to control.

The biosensor is also affected by chemical defects and mechanical impairments. The oxygen vacancy defect yields a statistical error in the system [12]. The IGZO has an amorphous structure, and there is an uncertain number of spaces that are without oxygen, which affects the flow of charge and leads to sensor noise. The other chemical defect comes from enzyme immobilizations. The number of captured GOx enzymes on the surface varies for each sensor. The number of hydrogen ions produced varies accordingly. The uncertainty yields different sensor values for glucose solutions of the same concentration.

One mechanical problem is the non-ideal flow of the liquids; this induces sporadic vibrations and appears in the output curve as abnormal spikes. In a use-case scenario, a sensor implanted beneath the human skin is inevitably exposed to physical noise. The other mechanical factor is a sensorial delay for symbol time synchronization, due to manual typing input. While the input is given manually, the time of release per each time slot is not perfectly matched with the ideal points. The traditional solution to such a problem is to embed a convolutional sliding window [14], but this method requires an impulse response and the linear effect of Inter-Symbol Interference (ISI).

In conclusion, the chip reactiveness varies for each sample, and a single chip has a short lifetime due to chemical uncertainties. The output curve is also exposed to the noise of both the time and the sensor value, due to mechanical instability. Data-detection algorithms should detect the sensor characteristics as quickly as possible and estimate the transmitted message, while distinguishing them from nonlinear noise. While considering well-known traditional solutions, it is prudent to consider a machine learning-based approach.

III. Detection Algorithms and Machine Intelligence

In this section, we introduce the communication feasibility by proposing data-detection algorithms that run on the testbed. The algorithms in the first subsection include a communica-

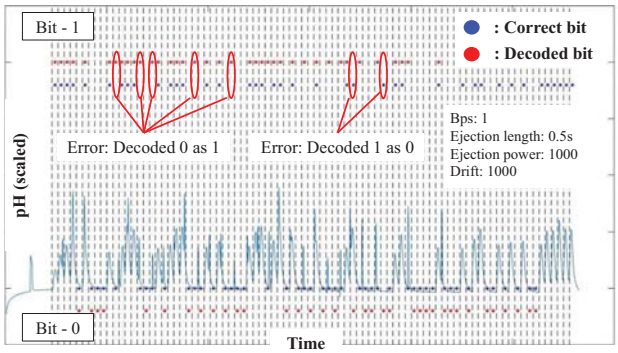

Fig. 3. A sample sequence response and decoding results of *naïve successive thresholding* and *spike classifying*.

tion model and baseline algorithms inspired by the testbed measurements. The second subsection describes how machine intelligence can enhance the data-detection performance.

A. Baseline Detection Algorithms

The data for the detection algorithms in this paper are a sequence of binary bits denoted by the vector $\mathbf{x}$. Let $\mathbf{y} = f(\mathbf{x})$, a vector, be a consequent sequence of probed data at the receiver, where f is a mapping from vectors to vectors with free lengths. The goal of the detection algorithm is to find a mapping $\hat{\mathbf{x}} = \mathcal{A}(f(\mathbf{x}))$ which is close to the inverse of f, such that the Hamming distance between $\hat{\mathbf{x}}$ and $\mathbf{x}$ is minimized.

A bitwise detector compartmentalizes the response vector $\mathbf{y}$ into bitwise response vectors $\mathbf{y}_1, \cdots, \mathbf{y}_l$, and the algorithm maps each $\mathbf{y}_i$ to $\hat{\mathbf{x}}_i$, where l is the number of bits. Synchronization between the transceivers is not secure; the response $\mathbf{y}$ contains both an idle part and a reactive part. The header bit protocol finds the starting point of the reactive part, and $\mathbf{y}_1$ starts from the detected starting point and ends at the beginning of the next time slot. Since the probing frequency is 8Hz, we have a size of $8t_s$-by-1 vector $\mathbf{y}_i$, where t_s is the symbol time length.

In Fig. III-A, we display a response to the sample input sequence and the detection results of proposed algorithms. The sample sequence consists of five header bits and 81 random bits. Vertical dotted lines represent the bitwise segmented response vector. The position of the blue dots in the interval indicates the input bit: bit-1 for the upper part, and bit-0 for the lower part. The red dots indicate the decoded bit. For example, when there is an only one red dot on the upper part of the interval, this means that the blue dot should be on the lower part, the input bit is 0, and the decoded bit is 1. In Fig. III-A, red dots indicate bits decoded by the *spike classifying* detection.

1) Naive Successive Thresholding: Let $\mathcal{A}_{nst}$ denote the first algorithm, called *naïve successive thresholding*. This algorithm maps $\mathbf{y}_i$ to $\hat{\mathbf{x}}_i$, by checking if a processed value from $\mathbf{y}_i$ is larger than a certain fixed threshold. This algorithm maps $\mathbf{y}_i$ to 1 if the maximum value of $\mathbf{y}_i$ is greater than the first value of $\mathbf{y}_i$ by more than a certain ratio of the value from pilot signals. This is given below in (2):

$$\boldsymbol{B}\left(\max\left(\mathbf{y}_i\right) - \mathbf{y}_i(1) > \frac{\eta_{nst}}{4} \cdot \sum_{j=1}^{4}\left(\max\left(\mathbf{y}_j\right) - \mathbf{y}_j(1)\right)\right), \tag{2}$$

where $\boldsymbol{B}$ is a Boolean function that returns 1 for true input status, and 0 otherwise, and $\mathbf{y}(i)$ denotes the i-th element of a vector $\mathbf{y}$. Note that the pilot signal is a reponse of the header bits $\{11110\}$, the denominator of 4 implies the averaging of first four bit-1 responses, and the threshold ratio η_{nst} is chosen empirically to minimize the average error for the training sets of measurements. This average may be subtracted from any input vector to obtain the relative response to the previous responses. The error rate on the sample sequence is calculated to be 23 out of 86.

The pilot sequence resolves the chemical problems by obtaining a threshold proportional to the pilot responses, but the mechanical problems still yield errors. The optimal threshold varies by sensor, but prior to communication, the threshold should be obtained using the practical decision algorithm. We choose the manual threshold that minimizes the average error everytime, where the performance guarantee is not secured for any other sets of chips. The more data we have, the more confident we are that the threshold is feasible.

2) Spike Classifying: In order to ameliorate any sporadic spike problems, we now consider a *spike classifying* algorithm. The first intuitive solution is to smooth out the peaks by introducing averaging windows. This solution turns out to be improper for the cases when the peak values are too high when windowed, and can severely affect neighboring values. The *spike classifying* algorithm classifies an interval vector $\mathbf{y}_{i:j} = (\mathbf{y}(i), \mathbf{y}(i+1), \cdots, \mathbf{y}(j))$ as a spike response or a proper reaction. If the interval is classified as a spike, the algorithm linearly interpolates the values and updates the response vector to $\bar{\mathbf{y}}$. The final output of the *spike classifying* is $\mathcal{A}_{nst}(\bar{\mathbf{y}})$.

The classification rule depends on its sharpness and height. The sharpness is defined by the length of an interval vector $\mathbf{y}_{i:j}$. Here the closed interval is taken such that the length is smaller than a symbol interval and the end values are $\mathbf{y}(i)$ and $\mathbf{y}(j)$, while having medium values greater than that. The height of the peak is the difference between the maximum value and the end values of the interval vector. The classifying threshold for those values is also manually taken from the observations. In the sample sequence, this algorithm reduces the number of errors in the previous algorithm by nine. Note that the thresholds rely on empirical search; hence, the credibility of the algorithm may increase with more observations.

B. Machine Intelligence-aided Detection

A machine intelligence-aided algorithm is an algorithm that improves itself by processing more data or data of higher quality. We explore how machine intelligence can enhance

a system of algorithm designs. To cope with varying input sizes and sequential data, we first suggest a *Vanilla Recurrent Neural Network (RNN)* as introduced in [13]. Then we enhance our module and switch-based machine learning algorithm, which selectively trains the features while economizing the training resources.

1) Vanilla RNN: The first issue is that the communication data are given sequentially and the total length of the input changes according to the chip. RNN is a well-known approach for such a case [7], which takes into account the previous decisions. This approach is distinct from successive detection by reflecting not only the previous decision but also the whole history of the decisions.

The *Vanilla RNN* assumes that a given environment can be modeled with a Hidden Markov Model (HMM), such that a Markov chain is constructed by the hidden states according to the time, and a correlation is yielded between the output states and the uncorrelated input states. The RNN is a combination of three independent networks, which are mappings from the input to a hidden state, a hidden state to the output, and from each hidden state to the next hidden state, denoted by $\mathbf{V}$, $\mathbf{U}$, $\mathbf{W}$, respectively.

The same structure can be used in two ways: to model a channel function $f : x \rightarrow \mathbf{y}$, or to find an inverse mapping $\mathcal{A} : \mathbf{y} \rightarrow \hat{\mathbf{x}}$. The former task can be carried out by using transmitted bits for the input states and respective bitwise responses for the output. In such a case, the approximated mapping is formulated as (3) with weight vectors $\mathbf{h}_i$, $\mathbf{b}_y$, $\mathbf{b}_h$.

$$\begin{aligned} \mathbf{h}_t &= f_h\left(\mathbf{V}x_t + \mathbf{W}\mathbf{h}_{t-1} + \mathbf{b}_h\right), \\ \mathbf{y}_t &= f_y\left(\mathbf{U}\mathbf{h}_t + \mathbf{b}_y\right). \end{aligned} \tag{3}$$

A detection algorithm via RNN can be constructed by training the weights with inversed input and output.

2) Module-based Universal Decoder: The problem of employing *Vanilla RNN* in our setup is that the number of memory tabs that influence the detection is less than the number of the usual sequential data. This is because when an input of bit-1 occurs, a syringe pumps the additional liquids into a thin pipe. The injected liquids push the existing medium onto the sensor and yield a washing effect. The delayed chemical reaction may induce an ISI bias only for bit-1.

We specify error-inducing three factors and solutions. The first one *naïve successive thresholding* deals with prevalent noise. Note that we take *spike classifying* as a preprocessing state so that physical vibration can be ignored. The second factor is the delayed reactions of the remaining chemicals on the chip surface, especially when bit-0 comes after a single input or multiple inputs of bit-1. This issue concerns the long memory of interference, which is partially addressed in *Vanilla RNN*. The third factor is the one-tab ISI, which observes the correlation between adjacent bitwise responses.

It is impractical to try to solve all of the above problems at once, as doing so would require too many resources. The problems do not affect the system simultaneously, and we do not even need to exploit all the solutions every time. On the other hand, if we ignore some of the problems for training,

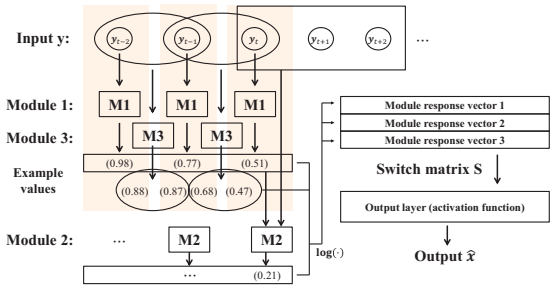

Fig. 4. A diagram of the module-based algorithm. Each of M1, M2, and M3 indicates the learning structure including weight matrices.

the variation of the error-rate values for the measurements increases over-training, while the mean value decreases. This implies that the achieved solution is unreliable in general.

The key idea of the algorithm is to separate the problems and to address them individually with machine learning. We use the term modules to refer to the individual machines. These modules can be selectively applied and integrated into the whole machine. At the beginning of the training, every module is turned on, and they are allocated the same switch values. When the training epoch advances, the main machine updates the switch values. As the training epoch advances, the main machine enhances the performance by updating the switch values, based on the contribution of each module. When the value reduces to a certain threshold, the switch turns off and discards the respective module. According to this process, the entire machine is trained to find which combination of modules would work for a given chip response. The machine observes idle state responses and the short pilot responses. The machines inside the modules help each module to find the optimal behaviors.

The first module consists of a simple bitwise artificial neural netowrk training algorithm that is almost identical to the *naïve successive thresholding*, where we take one more step before the final activation function. The final values indicate the certainty of the decision. As we have other modules to be combined, the output of the first module is a vector, where the $i-th$ element shows the probability that $\hat{\mathbf{x}}_i = 1$.

The second module to address the chemical delay uses the decision from the first module and concerns with how many consecutive bit-1s are likely to have resulted after the activation function. The more bit-1s that have been detected, the longer the time period of interest. This module takes a longer vector of $\mathbf{y'}_t = (\mathbf{y}_t, \mathbf{y}_{t+1}, \mathbf{y}_{t+2})$ as an input, and the output is the probability of denying the decision that $\hat{\mathbf{x}}_i = 1$, to reduce false positives.

The third module is also a simple neural network. However, it takes two consecutive responses as inputs to address one-tab ISI and yields as an output two consecutive bit decisions. In a similar way to the first module, the output is given by a vector of probabilities of bit-1 decisions. Note that other than the first and the last bits, each bit has two projected probabilities, and we choose the geometric mean of the two

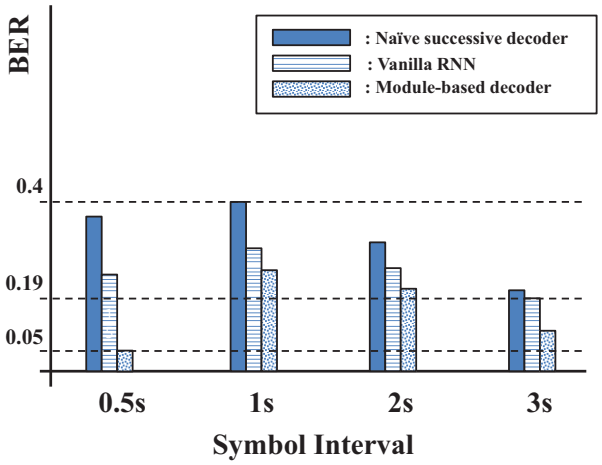

Fig. 5. A histogram of BER performances for the combination of three algorithms and four data rates.

values. This resembles the training of the $\mathbf{W}$ matrix in (3), where the difference is that the training range includes the whole system, other than the virtually hidden states.

Finally, we add a single layer to combine the decision values from the modules. The activation function maps the layer outputs to binary decisions. Fig. 4 describes the effective range of the input and the output for each module.

IV. Results

The measurements were taken for two weeks, where the two sets of biosensors are used for one week each. There were five to seven biosensors per a round of fabrication, and each chip lasts for a day. No significant behavior was observed between the sets.

For each symbol interval t_s of $\{0.5, 1, 2, 3\}$, the numbers of bits transmitted are (500, 780, 654, 191). Note that due to practical issues such as early retirement of the sensor or the premature running out of the messenger molecules, some of the measurement results have been discarded.

Fig. 5 shows the data detection results with the proposed algorithms for the data. The first algorithm, *naïve successive decoder*, is to show reasonable baseline performances. Noting that a random guess for the unbiased binary inputs yields an accuracy of 0.5, a naïve decoder that shows an accuracy of 0.4 for $t_s = 1$ and $t_s = 0.5$ is difference in considered ineffective. As shown in Fig. 5, the algorithm shows performance enhancements for increasing the symbol interval.

The module-based decoder shows remarkable improvement for $t_s = 0.5$, while it enhances the performance for all the other cases as well. It seems counter-intuitive that the faster data rate results in a better performance, with the same algorithm. The OOK modulation adds liquid into the system for bit-1, and the channel is narrow, meaning that the liquid injection pushes the remaining chemicals away. The faster repetition of injections removes the chemical more clearly and results in system steadiness. There is a trade-off between the washing effect and inter-symbol interference, as 1bps represents the worst performances for every algorithm.

In conclusion, the machine learning-aided algorithms have been proven here to enhance system performance. The module-based algorithm has been proposed to address the problems that occur for smaller sizes of training sets, and the algorithm has resulted in even greater improvements. The interesting part of the nanomachine is the trade-off between the washing and interference effects; this gives rise to the fastest data rate, along with the greatest accuracy.

V. Conclusions

In this paper, we proposed a nanomachine communication system that can communicate with itself inside a human blood vessel. We verified the concept by implementing the actual testbed and demonstrating bit sequences transmission. The first step of the testbed implementation was to design a nano-biosensor that was biocompatible, easily reproducible, and sufficiently small. For the sensor material, we exploited IGZO-based EGFET, which could detect glucose molecules using immobilized GOx on the chip surface. The convergence of a nanomachine and the communication task necessarily gives rise to several difficulties when building a communication protocol. We applied machine learning-based techniques to enhance the data-detection performance. This demonstration showed that communication was feasible via the testbed. With more data and further studies, future work may highlight more potential contained in nanomachines.

Acknowledgment

This research was supported in part by the Basic Science Research Program (NRF-2017R1A1A1A05001439), funded by the MSIT(Ministry of Science and ICT), through the National Research Foundation of Korea, and in part by the MSIT, under the "ICT Consilience Creative Program" (IITP-2019-2017-0-01015) supervised by the IITP.

References

[1] N. Farsad, H. B. Yilmaz, A. Eckford, C.-B. Chae, and W. Guo, "A comprehensive survey of recent advancements in molecular communication," *IEEE Commun. Surveys & Tut.,* vol.18, pp. 1887–1919, 2016.

[2] T. Nakano, T. Suda, M. Moore, R. Egashira, A. Enomoto, and K. Arima, "Molecular communication for nanomachines using intercellular calcium signaling," in *Proc. IEEE Conf. on Nanotechnology*, 2005.

[3] I. Llatser, D. Demiray, A. C.-Aparicio, D. T. Altilar, and E. Alarcon, "N3Sim: Simulation framework for diffusion-based molecular communication nanonetworks," *Elsevier Simul. Model. Prac. and Theory,* vol.42, pp. 210–222, 2014.

[4] N. Farsad, W. Guo, and A. W. Eckford, "Tabletop molecular communication: Text messages through chemical signals," *PLoS ONE*, vol.8, 2013.

[5] B.-H. Koo, C. Lee, H. B. Yilmaz, N. Farsad, A. Eckford, and C.-B. Chae, "Molecular MIMO: From theory to prototype," *IEEE J. on Sel. Areas Commun.,* vol.34, pp. 600–614, 2016.

[6] U. Rogers and M. S. Koh, "Parallel molecular distributed detection with Brownian motion," *IEEE Trans. Nanobioscience,* vol.15, pp. 871–880, 2016.

[7] N. Farsad, D. Pan, and A. Goldsmith, "A novel experimental platform for in-vessel multi-chemical molecular communications," in *Proc. IEEE GLOBECOM,* 2017.

[8] H. Unterweger, J. Kirchner, W. Wicke, A. Ahmadzadeh, D. Ahmed, V. Jamali, C. Alexiou, G. Fischer, and R. Schober "Experimental molecular communication testbed based on magnetic nanoparticles in duct flow," in *Proc. IEEE Int. Workshop on Sig. Proc. Adv. in Wireless Comm.,* 2018.
[9] Y. C. Shen, C. H. Yang, S. W. Chen, S. H. Wu, T. L. Yang, and J. J. Huang, "IGZO thin film transistor biosensors functionalized with ZnO nanorods and antibodies," *Biosens. Bioelectron.,* vol.54, pp. 306–310, 2014.
[10] J. Y. Hwang, M. T. Hong, E. J. Yun, and B. S. Bae, "Analog-to-digital converter with oxide thin-film transistors," *J. Inf. Disp.,* vol.17, pp. 79–85, 2016.
[11] B.-K. Sohn, B.-W. Cho, C.-S. Kim, and D.-H. Kwon, "ISFET glucose and sucrose sensors by using platinum electrode and photo-crosslinkable polymers," *Elsevier Sens. and Actuators B: Chem.,* vol.41, pp. 7–11, 1997.
[12] H.-K. Noh, K. J. Chang, B. Ryu, and W.-J. Lee, "Electronic structure of oxygen-vacancy defects in amorphous In-Ga-Zn-O semiconductors," *Physical Review B,* vol.84, 2011.
[13] N. Farsad, and A. Goldsmith "Neural network detection of data sequences in communication systems," in *IEEE Trans. on Sig. Proc.,* 2018.
[14] A. Noel and A. W. Eckford, "Asynchronous peak detection for demodulation in molecular communication," in *Proc. IEEE Int. Conf. on Commun.,* 2017.